High-Order Kinetic Relaxation Schemes as High-Accuracy Poisson Solvers

M. Mendoza
ETH Zürich, Computational Physics for Engineering Materials, Institute for Building Materials, Wolfgang-Pauli-Str. 27, HIT, CH-8093 Zürich, Switzerland.
mmendoza@ethz.ch

S. Succi
Istituto per le Applicazioni del Calcolo C.N.R., Via dei Taurini, 19 00185, Rome, Italy.
succi@iac.cnr.it

H. J. Herrmann
ETH Zürich, Computational Physics for Engineering Materials, Institute for Building Materials, Schafmattstrasse 6, HIF, CH-8093 Zürich, Switzerland.
hjherrmann@ethz.ch

We present a new approach to find accurate solutions to the Poisson equation, as obtained from the steady-state limit of a diffusion equation with strong source terms. For this purpose, we start from Boltzmann's kinetic theory and investigate the influence of higher order terms on the resulting macroscopic equations. By performing an appropriate expansion of the equilibrium distribution, we provide a method to remove the unnecessary terms up to a desired order and show that it is possible to find, with high level of accuracy, the steady-state solution of the diffusion equation for sizeable Knudsen numbers. In order to test our kinetic approach, we discretise the Boltzmann equation and solve the Poisson equation, spending up to six order of magnitude less computational time for a given precision than standard lattice Boltzmann methods.

Keywords: High Knudsen number, higher-order moments, diffusion equation, Poisson equation, lattice Boltzmann

1. Introduction

The diffusion equation is widely used to describe transport phenomena in which heat, mass, and other physical quantities are transferred in space and time due to underlying molecular collision processes [1]. At steady-state and when the source term does not depend explicitly on the concentration of the associated field, such diffusion processes settle down in the form of a non-local relation between the spatial distribution of the source and the resulting concentration of the associated field, a relation which is governed by the Poisson equation [2]. The notion of the Poisson equation as the steady-state solution of the diffusion equation extends to many other phenomena, e.g. electrostatics systems [3], plasma physics [4], chemistry and biology [5,6], molecular biophysics [7], density functional theory and solid-state physics [8,9,10], to name but a few. In fact, solving the Poisson equation accurately and efficiently continues to be a subject of intense research to this day.

From a microscopic point of view, the diffusion equation emerges from the underlying Boltzmann kinetic equation in the limit of very small Knudsen numbers, i.e. molecular mean free path much shorter than the typical macroscopic scale. Solving the Boltzmann kinetic equation to study diffusion processes makes apparently little sense, since the Boltzmann distribution function lives in a double-dimensional phase-space, i.e. position and velocity. However, in the last two decades, minimal (lattice) versions of the Boltzmann equation have been developed, in which velocity space is reduced to a small set of discrete velocities, so that the computational cost is cut down dramatically, making the solution of the lattice kinetic equation often competitive towards standard grid-discretisation of the corresponding partial differential equation.

The Lattice Boltzmann Method (LBM) has attracted much interest for solving the Navier-Stokes equations [11,12], and has been extended to solve other type of phenomena, e.g. diffusion equation [13,14], Maxwell equations[15], quantum systems [16], relativistic fluid dynamics [17,18], and the Poisson equation [2,19,20,21,22], to name but a few. To the best of our knowledge, to date, the solution of the Poisson equation based on kinetic theory has been confined to smooth (non-stiff) source terms and small Knudsen numbers [23]. If the target is the steady-state solution of the diffusion equation, as it is the case for the Poisson equation, the kinetic approach looses much of its appeal, since its inherently real-time dynamics must proceed through smaller time-steps than those affordable by fictitious-time iteration techniques. In this paper we shall show that such gap can be closed by resorting to higher-order kinetic formulations, at least for the cases where the source term does not depend on the concentration field avoiding artificial solutions as pointed out in Ref. [2].

Higher-order formulations of the LB equation have been developed before, mostly in the context of thermal [24] and relativistic fluid dynamics [25], and recently, density functional theory [26]. There, the main idea is to design the equilibrium distribution in such a way to include not only the conserved moments, such as density and current, but also the non-conserved ones, namely the momentum flux tensor and the heat flux, and eventually higher order kinetic moments with no direct hydrodynamic significance (sometimes called "ghosts"). Clearly, in order to match this longer ladder of kinetic moments, a correspondingly larger set of discrete velocities is required. It should be emphasized that the discrete kinetic equation is a superset of the diffusion equation, and consequently, to the purpose of solving the diffusion equation alone, higher-order contributions must be regarded as discretisation artefacts which need to be minimised, and possibly canceled altogether. This is precisely the target of this paper. Here, we will use higher-order lattices and associated equilibria, to remove higher-order contributions to the macroscopic equations reproduced by the Boltzmann equation. For this purpose, we develop the respective LBM and show that, by expanding the equilibrium distribution up to seventh order in Hermite polynomials, one can solve the Poisson equation *six orders* of magnitude faster than with standard LB models, at a given level accuracy.

The paper is divided as follows: In Sec. 2, we introduce the diffusion equation with stiff source term and the associated Boltzmann kinetic equation. In Sec. 3, we describe the influence of higher order moments on the steady-state solution of the diffusion equation. In Sec. 4, we develop a LBM for the Poisson equation, validate and compare the model with standard cases. Finally, we discuss the results and outlook of our work.

2. From Boltzmann kinetic theory to diffusion equation

Let us begin by writing the diffusion equation for a scalar Φ in the standard form:

$$\frac{\partial \Phi}{\partial t} = D\nabla^2 \Phi + S \quad , \tag{1}$$

where Φ is the concentration of particles (temperature in the case of the heat transfer equation), D is the diffusivity, and S is the source term, which, in our case, depending neither on Φ nor on time. The steady-state version of Eq. (1) delivers the Poisson equation with source $-S/D$. It is well-known that this equation can be obtained from an underlying Boltzmann kinetic equation, via a Chapman-Enskog expansion, in the limit of small Knudsen numbers, $Kn = \lambda/L$, being λ the mean-free path and L the characteristic size of the system [27].

The Boltzmann equation in the single relaxation time approximation (BGK) [28] reads as follows:

$$\frac{\partial f}{\partial t} + \vec{v} \cdot \nabla f = -\frac{1}{\tau}(f - f^{\mathrm{eq}}) + \mathcal{S} \quad , \tag{2}$$

where $f \equiv f(\vec{x}, \vec{v}, t)$ is the probability distribution function, $\vec{v}$ are the microscopic velocity vectors, τ is the relaxation time, and f^{eq} is the equilibrium distribution.

In the non-relativistic context, the local equilibrium is given by a Maxwell-Boltzmann distribution:

$$f^{eq} = \frac{e^{-(v-u)^2/2c_s^2}}{(2\pi c_s^2)^{3/2}} \quad , \tag{3}$$

where u is the local flow speed. Such local equilibrium encodes the basic symmetries of Newtonian mechanics, particularly Galilean invariance, which is built-in via the dependence on the molecular speed relative to the fluid $v - u$, rather than the absolute one, v. In order to secure Galilean invariance for *any* fluid speed, kinetic moments at *all orders* should be matched on the lattice. This would require an infinite connectivity, which is clearly unviable on any realistic lattice. As a result, LB local equilibria are typically based on finite-order Hermite expansions, of the form [29]:

$$f^{\mathrm{eq}}(\vec{x}, \vec{v}, t) = w(\vec{v}) \sum_{n=0}^{\infty} a_n^{(n)}(\vec{x}, t) H_n^{(n)}(\vec{v}) \quad , \tag{4}$$

where $H_n^{(n)}$ are the tensorial Hermite polynomials of order n, and $w(v)$ is the weight function defined as,

$$w(\vec{v}) = \frac{1}{(2\pi c_s^2)^{3/2}} e^{-v^2/2c_s^2} \quad , \tag{5}$$

c_s being the sound speed. Note that $w(v)$ is the uniform *global* equilibrium, corresponding the no-flow limit of the local equilibrium f^{eq}. The coefficients $a_n^{(n)}$ (kinetic moments) are calculated by projecting the equilibrium distribution onto the respective Hermite polynomial,

$$a_n^{(n)} = \int f^{\text{eq}} H_n^{(n)} \; d^3v \quad . \tag{6}$$

The lowest-order kinetic moments have a direct macroscopic interpretation. For instance, by integrating on the velocity space ($H_0 = 1$),

$$\Phi = \int f \; d^3v = \int f^{\text{eq}} \; d^3v \quad , \tag{7}$$

where the last equality comes from the requirement of mass conservation on the collsion operator, namely:

$$\frac{1}{\tau} \int (f - f^{\text{eq}}) \; d^3v = 0 \quad . \tag{8}$$

The diffusivity of the system is dictated by the relaxation time as

$$D = c_s^2 \tau. \tag{9}$$

It can be shown that the Boltzmann equation recovers the diffusion equation in the limit of small Knudsen numbers, which is related with the mean-free path and therefore with the relaxation time. This is the strongly-interacting fluid regime, whereby the particle dynamics is collision-dominated, so that collective behaviour sets in on a short timescale τ. The opposite limit of large Knudsen numbers, hence large values of τ, corresponds to the free-particle motion (ballistic regime), whereby memory of the initial conditions is kept for a very long time. It is therefore clear that in the ballistic regime higher order moments of the equilibrium distribution do not relax and consequently the Boltzmann equation does not converge to any standard diffusion equation. On the other hand, by increasing the relaxation time, hence the effective diffusivity, one would intuitively expect a quickest path to steady-state. In this respect, it would be highly desirable to derive a diffusion equation in kinetic form, which can attain steady-state solution at relatively large Knudsen numbers by exactly cancelling as many high-order terms as possible.

As discussed above, this can be achieved by zeroing as many higher order contributions as possible, while retaining the largest possible diffusion coefficient.

3. Influence of Higher-Order Terms

Let us consider the case when all time derivatives are zero, $\partial/\partial t = 0$. The Boltzmann equation, Eq. (2), can be written as,

$$f = f^{\rm eq} + \tau \mathcal{S} - \tau \vec{v} \cdot \nabla f \quad . \tag{10}$$

By integrating this equation in velocity space, we obtain a steady-state continuity equation for the current density:

$$\nabla \cdot \vec{J} = S \quad , \tag{11}$$

where $\vec{J} = \int f \vec{v}\, d^3v$ and $S = \int \mathcal{S}\, d^3v$.

Multiplying Eq. (10) by $\vec{v}$ and integrating again, we obtain

$$\vec{J} = \vec{J}^{\rm eq} + \tau \vec{S} - \tau \nabla \cdot \Pi \quad , \tag{12}$$

with $\Pi \equiv \Pi_{\alpha\beta} = \int f v_\alpha v_\beta\, d^3v$ being the momentum-flux tensor and $\vec{S} = \int \mathcal{S} \vec{v}\, d^3v$ the source current. The first term at the right-hand-side takes the form $\vec{J}^{eq} = \Phi \vec{u}$, where $\vec{u}$ is the flow speed in the local equilibrium. For the case of a purely diffusive dynamics this term is zero. The second term contributes a shift $\tau \vec{S}$ to the flow speed, and it must also be set to zero for the case of pure diffusion. Finally, the pressure tensor shoud reduce to $c_s^2 \Phi$, so that inserting (12) into (11), the diffusion equation is obtained. The first two conditions are automatically ensured by setting $\vec{u} = 0$ in the local equilibrium. The third one, however, cannot be enforced exactly because of higher order contributions to the momemtum flux tensors.

Indeed, by iterating the procedure, one obtains the general expression:

$$\sum_{n=0}^{\infty} (-1)^n \tau^n \nabla^{n+1} \left[\Pi^{{\rm eq}(n+1)} + \tau S^{(n+1)} \right] = S \quad , \tag{13}$$

where $\Pi^{{\rm eq}(n)} \equiv \int f^{\rm eq} v_{\alpha_1} ... v_{\alpha_n}\, d^3v$ and $S^{(n)} \equiv \int \mathcal{S} v_{\alpha_1} ... v_{\alpha_n}\, d^3v$, are both tensors of order n. Note that for small Knudsen numbers ($\tau \nabla \ll 1$) higher-order terms become negligible and the series is convergent. We also observe that the solution is dictated by the equilibrium distribution and the source term, so that, by properly choosing both expressions, higher order terms can be canceled thus opening the way to high-accuracy solutions of the Poisson equation.

In numerical practice, accuracy is typically improved by increasing the resolution of the grid. This imposes correspondingly smaller time steps, which is clearly expensive, especially in three dimensions. High-order methods are meant to mitigate the problem by achieving the same level of accuracy with many less grid points. They are typically based on higher-order stencils for the discretisation of the corresponding differential operators.

In this work, we present a procedure whereby the same goal is achieved by drawing directly from an underlying lattice kinetic theory, i.e. by a suitable (non-Gaussian) extension of the local kinetic equilibria as combined with the use of larger sets of discrete speeds than those typically employed in standard Lattice Boltzmann theory.

4. Kinetic approach to the Poisson equation

In order for the Boltzmann kinetics to converge to the Poisson equation, the following conditions need to be met:

$$\Pi^{\mathrm{eq}(1)} = \Pi^{\mathrm{eq}(n>2)} = 0, \qquad (14)$$
$$S^{(n>0)} = 0.$$

The equilibrium distribution can be expressed as a separable product of a function of the microscopic velocity (global uniform equilibrium) and a function of the time-spatial coordinates, namely:

$$f^{\mathrm{eq}} = \phi(\vec{v})\Phi(\vec{x},t) \quad , \qquad (15)$$

and similarly for the source term:

$$\mathcal{S} = \chi(\vec{v})S(\vec{x},t) \quad . \qquad (16)$$

In the above, $\chi(\vec{v})$ and $\phi(\vec{v})$ are velocity dependent functions that need to be determined based on the conditions (14) above. To this purpose, we write:

$$\phi(\vec{v}) = w(\vec{v})\sum_{n=0}^{N} b_n^{(n)} H_n^{(n)}(\vec{v}) \quad , \qquad (17)$$

and,

$$\chi(\vec{v}) = w(\vec{v})\sum_{n=0}^{N} c_n^{(n)} H_n^{(n)}(\vec{v}) \quad , \qquad (18)$$

where $b_n^{(n)}$ and $c_n^{(n)}$ are constant coefficients and N is a cut-off truncating the Hermite expansion. With reference to the one-dimensional case and an expansion up to seventh order ($N=7$), we obtain:

$$\phi(v) \simeq w(v)\left(1 - \frac{3c_s^2 - 6c_s^2 v^2 + v^4}{8c_s^4} - \frac{15c_s^6 - 45c_s^4 v^2 + 15c_s^2 v^4 - v^6}{24c_s^6}\right) \quad , \qquad (19)$$

and

$$\chi(v) \simeq w(v)\left(1 + \frac{1}{2}\left[1 - \frac{v^2}{c_s^2}\right] + \frac{3c_s^2 - 6c_s^2 v^2 + v^4}{8c_s^4} + \frac{15c_s^6 - 45c_s^4 v^2 + 15c_s^2 v^4 - v^6}{48c_s^6}\right), \qquad (20)$$

where the weight $w(v)$ for the one-dimensional case is defined as

$$w(v) = \frac{1}{(2\pi c_s^2)^{1/2}} e^{-v^2/2c_s^2} \quad . \qquad (21)$$

With Eqs. (19) and (20), one can prove that the the following expressions for the moments are fulfilled: $\Pi^{\mathrm{eq}(0)} = \Phi$, $\Pi^{\mathrm{eq}(1)} = 0$, $\Pi^{\mathrm{eq}(2)} = \Phi c_s^2$, $\Pi^{\mathrm{eq}(n>2)} = 0$, $S^{(0)} = S$, $S^{(n>0)} = 0$. These expressions hold only up to $N=7$.

To develop a lattice Boltzmann model and solve the Poisson equation numerically, we need to impose a quadrature and find a corresponding set of finite velocity vectors, v_i, such that the orthogonality conditions are fulfilled to the desired order [24,25], namely:

$$\int w(v) H_n(v) H_m(v)\, dv = \sum_{i=0}^{N_v} w_i H_n(v_i) H_m(v_i) \quad . \tag{22}$$

Here w_i are discrete weights and H_n are the one-dimensional Hermite polynomials. The number of velocity vectors N_v depends on the order of the expansion that we intend to reproduce with the quadrature. For expansions up to seventh order, we need at least 13 velocity vectors and weights. Each set of numbers also provides its own lattice sound speed c_s, which goes from about 0.4 with $N = 3$ to nearly 1.2 with $N = 7$. The details of these values are given in Appendix A.

Higher-order schemes are usually exposed to numerical instabilities, due to the appearance of additional modes in the dispersion relation. However, since our procedure is designed to annihilate precisely these modes, we expect our approach to be stable. Furthermore, the method proposed here can also be used as a systematic technique of calculating weights and velocity vectors, such that one can achieve the desired level of accuracy for the computation of Laplacian operators, similar to the work proposed in Refs. [30,31].

We have performed the theoretical analysis in one-dimension, however, extensions to two- and three-dimensions are straightforward, by using the tensorial form of the Hermite polynomials and by finding the discrete velocity vectors with the respective orthogonality conditions. For instance, in order to fulfill the conditions (14), up to fifth order, the following expansions apply:

$$\phi(\vec{v}) \simeq w(\vec{v}) \left(1 - \frac{15c_s^4 - 10c_s^2\vec{v}^2 + \vec{v}^4}{8c_s^4}\right) \quad , \tag{23}$$

and

$$\chi(\vec{v}) \simeq w(\vec{v}) \left(1 + \frac{1}{2}\left[3 - \frac{\vec{v}^2}{c_s^2}\right] + \frac{15c_s^4 - 10c_s^2\vec{v}^2 + \vec{v}^4}{8c_s^4}\right), \tag{24}$$

where the weight $w(\vec{v})$ is defined as

$$w(\vec{v}) = \frac{1}{(2\pi c_s^2)^{3/2}} e^{-\vec{v}^2/2c_s^2} \quad . \tag{25}$$

For the three-dimensional case, we should calculate the corresponding set of discrete velocity vectors, $\vec{v}_i$, such that the orthogonality condition is fulfilled up to fifth order, namely:

$$\int w(\vec{v}) H_m^{(m)}(\vec{v}) H_n^{(n)}(\vec{v}) d^3v = \sum_{i=0}^{N_v} w_i H_m^{(m)}(\vec{v}_i) H_n^{(n)}(\vec{v}_i), \tag{26}$$

which is the three-dimensional version of Eq. (22). By solving these algebraic equations, we find that we need at least 111 velocity vectors and 10 weights. Details are given in Appendix A.

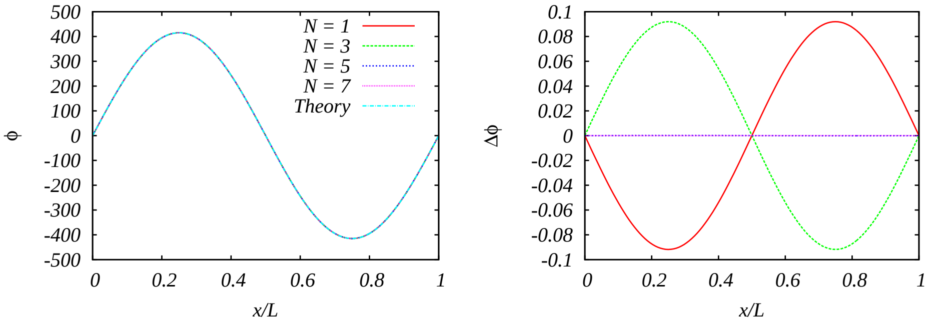

Fig. 1. Solution of the Poisson equation for different cut-off N and $l = 1$. The absolute error is computed $\Delta\phi \equiv \phi_n - \phi_t$, where ϕ_n is the potential calculated with the lattice Boltzmann model, and ϕ_t is the analytical solution of Eq. (27). The system size is $L = 128$ lattice cells.

5. Numerical Results

As an application, we solve the Poisson equation [3],

$$\nabla^2\Phi = -\frac{\rho}{\epsilon} \quad , \tag{27}$$

where ρ is the charge density and ϵ is the electric permittivity.

Replacing the equilibrium distribution f^{eq} from Eq. (15) into Eq. (13), we obtain:

$$S + c_s^2\tau\nabla^2\Phi \simeq 0 \quad , \tag{28}$$

implying that $S = \rho c_s^2\tau/\epsilon$. In the lattice Boltzmann model, one must take into account second order corrections to the relaxation time, $\tau = \tau_{lb} - 1/2$, where τ_{lb} the lattice Boltzmann relaxation time.

From Eq. (13), it is appreciated that the spatial derivatives of the source also introduce errors in the solution. Thus, by annihilating higher order contributions, we are also eliminating spurious effects due to the derivatives of the source term. If the source term is smooth, this derivatives are negligible, however, for stiffer source terms, one can use the present approach to add accuracy to the solution. In order to investigate this issue, we solve the potential for the following charge density:

$$\rho(x) = \epsilon \sin\left(\frac{2\pi x}{L} l\right) \quad , \tag{29}$$

where the integer l controls the smoothness of the derivatives of the charge density, and L is the simulated length, with $x \in [0, L)$. We use periodic boundary conditions for simplicity, and $\tau_{lb} = 1$ (numerical units).

We express the relative error as

$$\epsilon = aL^{-\alpha} \tag{30}$$

where the amplitude a and scaling exponent α (order of accuracy) both depend on N. From Fig. 1, we see that for a relatively large system size, $L = 128$, the first order expansion is sufficient to produce satisfactory results, with errors around 0.02% for

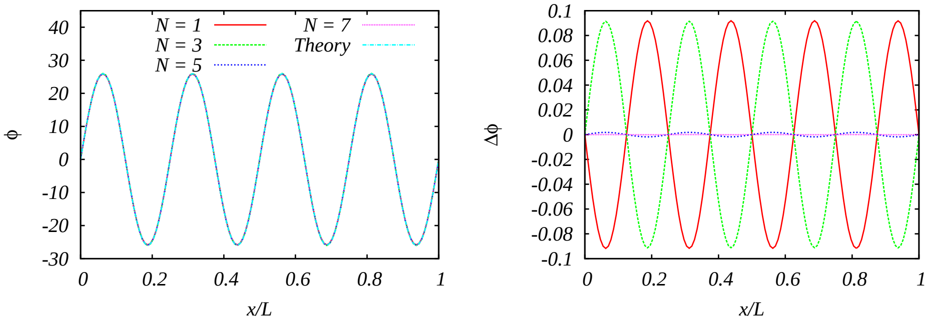

Fig. 2. Solution of the Poisson equation for different cut-off N and $l = 4$. The absolute error is computed $\Delta\phi \equiv \phi_n - \phi_t$, where ϕ_n is the potential calculated with the lattice Boltzmann model, and ϕ_t is the analytical solution of Eq. (27). The system size is $L = 128$ lattice cells.

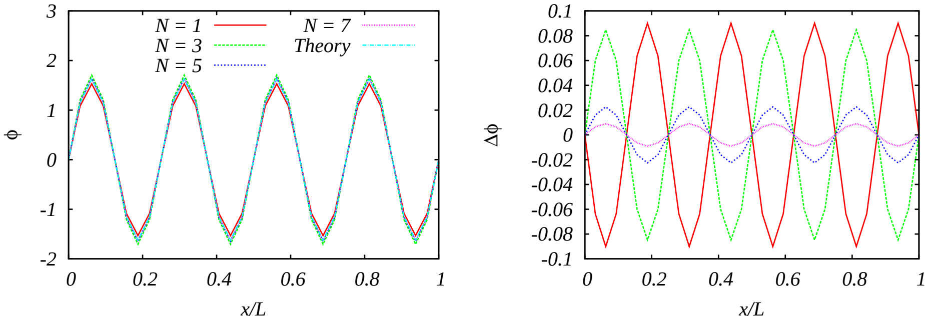

Fig. 3. Solution of the Poisson equation for different cut-off N and $l = 4$. The absolute error is computed $\Delta\phi \equiv \phi_n - \phi_t$, where ϕ_n is the potential calculated with the lattice Boltzmann model, and ϕ_t is the analytical solution of Eq. (27). The system size is $L = 32$ lattice cells.

$N = 1$ and $N = 3$, and below $10^{-3}\%$ for $N = 5$ and $N = 7$. In Fig. 2, we see that by increasing the derivatives of the charge density (increasing l), the error increases for all values of the cut-off N. However, as we have mentioned before, the effect of removing higher-order errors becomes crucial for the case of small system sizes. By decreasing the number of lattice cells (see Fig. 3), the discrepancies become visual and the errors are around 5% for $N = 1, 3$, 1% for $N = 5$, and less than 0.4% for $N = 7$.

Finally, by further reducing the lattice resolution, $L = 8$ cells, and considering only one oscillation, $l = 1$, we see that the errors for the first order expansion, $N = 1$, are around 7%, while for the highest values of $N = 7$ they remain below 0.5%. The fact that one can obtain very small errors by increasing the order of the expansion in the velocity space of the equilibrium distribution and source term, opens up the possibility of highly accurate solutions on very small grids.

Increasing the order of the expansion inevitably implies an overhead of numerical operations per time step, thereby slowing down the simulations. On the other hand,

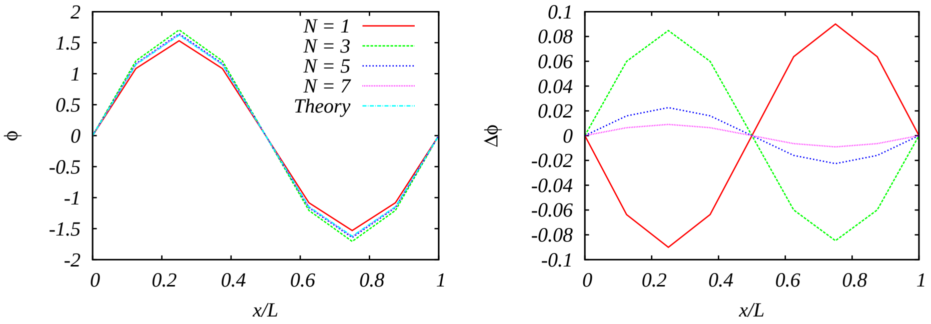

Fig. 4. Solution of the Poisson equation for different cut-off N and $l = 1$. The absolute error is computed $\Delta\phi \equiv \phi_n - \phi_t$, where ϕ_n is the potential calculated with the lattice Boltzmann model, and ϕ_t is the analytical solution of Eq. (27). The system size is $L = 8$ lattice cells.

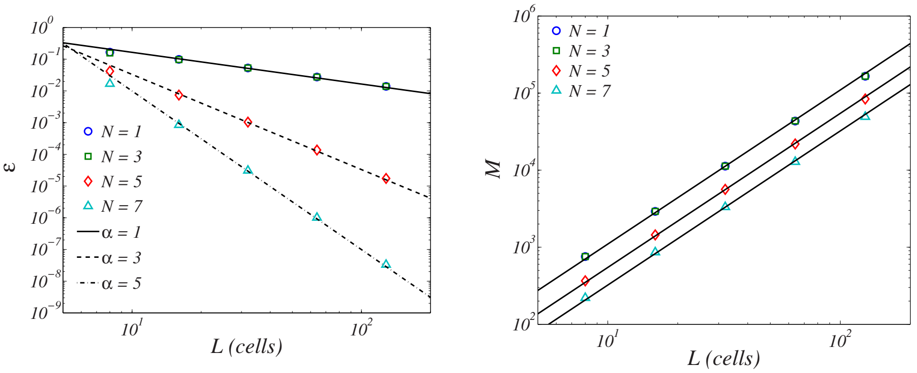

Fig. 5. (left) Relative error as a function of the system size L for each cut-off value N of the Hermite polynomials expansion. The lines denote the fitting curves using the general expression, $\epsilon = a_N L^{-\alpha}$ (right). Number of iterations as a function of the system size L for each cut-off value of N. The lines denote the fitting curve using the general expression, $M = b_N L^2$, M being the number of time-steps.

higher order lattices also bring aditional side-benefits such as a higher sound speed c_s, leading to a larger diffusion coefficient and, consequently, to a faster path to the steady-state solution. In order to highlight the concrete advantages of our approach, we next implement the same simulations for different orders N and inspect the CPU time, number of iterations, and system size, required to achieve a specific level of accuracy.

5.1. *Computational performance*

Let us next change the order of the expansion and analyse how the relative error decreases with the system size. From Fig. 5 (left), we see that, according to our expectations, by increasing the order of the expansion, the order of convergence,

Table 1. Parameters a_N and b_N for each value of N. Also shown are the accuracy exponent α_N and the number of lattice sites updated per second, ξ_N, in Msites/s (Million sites per second).

N	a_N	b_N	ξ_N (Msites/s)	ξ_N/b_N	α_N
1	1.645	10.98	26.2	2.39	1
3	1.645	10.98	21.4	1.95	1
5	33.11	5.46	5.44	0.997	3
7	992.3	3.23	5.76	1.78	5

measured by the exponent α, also increases and in a very substantial way. From the same figure (right), we can also observe that for all values of N, the number of time-steps M scales quadratically with the lattice size, and decreases with the order of the expansion N. The former feature is the typical signature of diffusion process, while the second reflects the fact that higher order lattice deliver a larger sound speed, hence they take less time-steps to achieve a given level of accuracy.

The computational time required to achieve a given accuracy clearly depends on the computational speed of the model on each given machine. This is conveniently expressed in terms of the number of lattice sites updates per CPU second, ξ_N. This quantity is expected to be a decreasing function of N, since higher-order quadratures require more operations per lattice site.

Actual measurements give the values reported in Table 1. Although no clear trend can be extracted from these measurements, it is observed that the cases $N = 5$ and $N = 7$ are about five times more expensive than the cases $N = 1$ and $N = 3$. Also to be noted that $\xi_7 > \xi_5$, which is counter-intuitive. However, upon inspecting the velocity vectors (see Appendix Appendix A), one can appreciate that for $N = 7$ the velocities are consecutive (no velocity gap), while for $N = 5$ they are not. Since, memory access is faster for consecutive elements, we expect the case $N = 7$ to perform better than $N = 5$. More generally, the dependence on N appears to be highly dependent on memory access patterns, which are not easily quantified through simple scaling relations.

The total CPU time, $t_{L,N}$, it takes to attain steady-state for a given system size L can be written as follows:

$$t_{L,N} = \frac{b_N}{\xi_N} L^3 \quad . \tag{31}$$

In the above, the cubic dependence derives from the L^2 diffusive scaling, as combined with the number of sites L (the space-time computational volume of a diffusive process scales like L^{2+D} in D spatial dimensions). Finally, b_N is a relative scale for the number of time-steps, which is expected to decrease at increasing N because of the increasing sound speed. The actual values of obtained in the simulations are reported in Table 1. As an example, for the case $L = 100$, we find that the simulations take 0.42 (65629 time steps), 0.51 (65629 time steps), 1.0 (33769 time steps), and 0.56 (20079 time steps) seconds, for $N = 1, 3, 5, 7$ respectively.

This shows that the total computational time remains within a factor two across

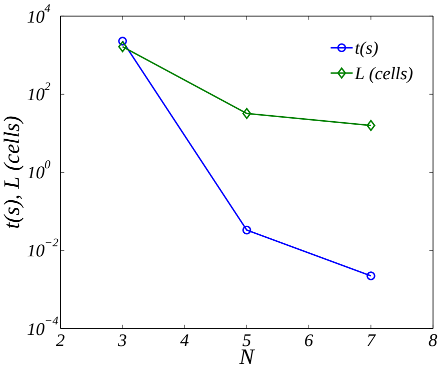

Fig. 6. Computational time t_N and system size L needed to recover the solution of the Poisson equation with a relative error of 0.1%, for different N.

all values of N, while the error with respect to the analytical solution decreases dramatically at increasing N (see Fig. 5). Hence, we can conclude that increasing N leads to a dramatic boost of accuracy at a very moderate extra computational cost. Finally, we have inspected the computational time to steady-state for a given accuracy, at changing the size of the problem and the order of the quadrature. By combining the previous relations, (30) and (31), we obtain:

$$t_N(\epsilon) = \frac{b_N}{\xi_N}\left(\frac{a_N}{\epsilon}\right)^{3/\alpha_N} \quad . \tag{32}$$

The plot reports the computational time and system size needed to achieve a prescribed relative error, for the case in point $\epsilon = 10^{-3}$. From fig. 6, we see that by increasing the order of the expansion in Hermite polynomials, it is possible to reduce the computational time by several orders of magnitude (up to six).

These data refer specifically to the solution of the Poisson equation, but we have reasons to believe that similar conclusions would hold for other partial differential equations with a kinetic-theory background, including many linear non-linear partial differential equations of great relevance in many branches of modern science.

5.1.1. *Comparison with multi-grid methods*

In order to gain information about the performance of this approach, as compared with existing tools to solve the Poisson equation, we have implemented simulations using the multi-grid (MG) method [32,33]. The number of nodes in the MG simulations is denoted by L, like for the lattice Boltzmann method. Fixing $l = 1$ (from Eq. (29)), in Table 2 we show the relative errors per node, ϵ/L, together with the corresponding computational time. Note that for $N = 3$, the MG method is faster than our approach, with the same second order accuracy. By increasing N, however, the MG becomes increasingly faster, but also increasingly less accurate at a given grid size L. For instance, in order to attain the same accuracy of the case $N = 7$ on

Table 2. Average relative error per node, ϵ/L, for the solution of the Poisson equation using different orders N and the multi-grid method (MG). Inside the parenthesis, we report the computational time in milliseconds.

L	$N=3$	$N=5$	$N=7$	MG
16	$10^{-2}(2)$	$9\times10^{-4}(2)$	$10^{-4}(2)$	$10^{-2}(<1)$
32	$3\times10^{-3}(4)$	$6\times10^{-5}(4)$	$2\times10^{-6}(6)$	$3\times10^{-3}(<1)$
64	$8\times10^{-4}(26)$	$4\times10^{-6}(29)$	$3\times10^{-8}(38)$	$8\times10^{-4}(<1)$
2^{14}	–	–	–	$2\times10^{-8}(12)$

$L = 64$, the MG solver requires $L = 16384$ grid points, being still faster by about a factor 3. Given that $N = 7$ requires 13 arrays, the LB memory saving is about a factor 20. These figures indicate that the present high-order LB Poisson solver is consistently slower than MG, however it also takes correspondingly less memory to achieve high levels of accuracy. Given that MG represents the state-of-the art for fast Poisson solvers, these can be regarded as satisfactory results.

In summary, we have shown that the present LB approach provides a viable option for the solution of Poisson equation, in the sense of simplicity and possibly also in terms of memory demand at high accuracy.

6. Conclusions

Summarizing, we have studied the effect of higher order terms on the steady-state solution of the diffusion equation, using a lattice version of the Boltzmann equation with high-order (non-gaussian) equilibria. For the numerical solution, we have developed a higher order lattice Boltzmann model capable of reproducing up to seventh order moment of the equilibrium distribution and source charge. This permits to cancel exactly error terms up to seventh order. For the validation and study of our approach, we have solved the one-dimensional Poisson equation, and found that the computational time to steady-state, at a prescribed level of accuracy, can be reduced by up to six orders of magnitude as compared to standard LB formulations. The six orders gain in computational time is due to the fact that higher order lattices permit to utilize much smaller grid sizes. Moreover, since they contain a large number of discrete velocities, they allow faster propagation within the lattice, hence a correspondingly faster attainement of the steady-state. We have also compared with state-of-the art multigrid solvers, and found that high-order LB is nearly competitive in efficiency at a lower memory demand.

Extensions to two and three-dimensions, as well as time-dependent diffusion and advection-diffusion equations will make the object of future research.

Acknowledgements

We acknowledge financial support from the European Research Council (ERC) Advanced Grant 319968-FlowCCS.

Appendix A. Quadrature of the Boltzmann Equation

Using Eq. (22) for each order of expansion N, we can calculate the velocity vectors v_i with the respective weights w_i. In addition, each set of w_i and v_i provides a characteristic normalised speed c_s, $c_s^2 = \sum_{i=0}^{M} w_i v_i^2$. Here we only show the cases for $N = 3, 5$, and 7, since the simulations for $N = 1$ were performed with the same lattice than for $N = 3$. For $N = 3$, we have

$$\begin{aligned}
w_0 &= 0.63664690312607816284434 61\\
w_1 &= 0.18141458774368577505004149\\
w_2 &= 0.18141458774368577505004149\\
w_3 &= 0.000261960693275143527785461 5\\
w_4 &= 0.000261960693275143527785461 5\\
v_0 &= 0, v_1 = 1, v_2 = -1, v_3 = 3, v_4 = -3\\
c_s^2 &= 1 - \sqrt{2/5} \quad ,
\end{aligned} \tag{A.1}$$

for $N = 5$,

$$\begin{aligned}
w_0 &= 0.45813515550767658573\\
w_1 &= 0.23734280857794043891\\
w_2 &= 0.23734280857794043891\\
w_3 &= 0.032324653788654934092\\
w_4 &= 0.032324653788654934092\\
w_5 &= 0.0012640621515365148385\\
w_6 &= 0.0012640621515365148385\\
w_7 &= 8.977280298192933351 \times 10^{-7}\\
w_8 &= 8.977280298192933351 \times 10^{-7}\\
v_0 &= 0, v_1 = 1, v_2 = -1, v_3 = 2, v_4 = -2,\\
v_5 &= 3, v_6 = -3, v_7 = 5, v_8 = -5\\
c_s^2 &= 0.86952909818721338884964163917209 \quad ,
\end{aligned} \tag{A.2}$$

Table 3. Velocity vectors $\vec{v}_i$ with their corresponding weights w_i for $N = 5$. There are 111 velocity vectors and 10 different weights.

w_i	$\vec{v}_i$
0.15014405	$(0,0,0)$
0.02500399	$(\pm1,0,0),(0,\pm1,0),(0,0,\pm1)$
0.04505812	$(\pm1,\pm1,0),(0,\pm1,\pm1),(\pm1,0,\pm1)$
$7.81706951 \times 10^{-7}$	$(\pm3,\pm3,\pm3)$
$9.79290909 \times 10^{-6}$	$(\pm3,\pm3,0),(\pm3,0,\pm3),(0,\pm3,\pm3)$
$1.81207346 \times 10^{-4}$	$(\pm3,\pm1,0),(\pm3,0,\pm1),(0,\pm1,\pm3)$ $(\pm1,\pm3,0),(\pm1,0,\pm3),(0,\pm3,\pm1)$
$6.03109965 \times 10^{-4}$	$(\pm2,\pm2,0),(\pm2,0,\pm2),(0,\pm2,\pm2)$
$3.49417975 \times 10^{-3}$	$(\pm2,\pm1,\pm1),(\pm1,\pm1,\pm2),(\pm1,\pm2,\pm1)$
$1.05490305 \times 10^{-2}$	$(\pm2,0,0),(0,\pm2,0),(0,0,\pm2)$
$4.50016852 \times 10^{-5}$	$(\pm3,0,0),(0,\pm3,0),(0,0,\pm3)$

and finally, for $N = 7$,

$$\begin{aligned}
w_0 &= 0.3455934552621565 \\
w_1 &= 0.2374599218260301 \\
w_2 &= 0.2374599218260301 \\
w_3 &= 0.07705730993964580 \\
w_4 &= 0.07705730993964580 \\
w_5 &= 0.011801423732312036 \\
w_6 &= 0.011801423732312036 \\
w_7 &= 0.00085524660095l3439 \\
w_8 &= 0.0008552466009513439 \\
w_9 &= 0.00002884934614927074 \\
w_{10} &= 0.00002884934614927074 \\
w_{11} &= 5.209238332209471 \times 10^{-7} \\
w_{12} &= 5.209238332209471 \times 10^{-7} \\
v_0 &= 0, v_1 = 1, v_2 = -1, v_3 = 2, v_4 = -2 \\
v_5 &= 3, v_6 = -3, v_7 = 4, v_8 = -4, v_9 = 5 \\
v_{10} &= -5, v_{11} = 6, v_{12} = -6 \\
c_s^2 &= 1.1544053947399681272395977588380 \quad .
\end{aligned} \tag{A.3}$$

For the three dimensional case, up to fifth order, we can find the velocity vectors $\vec{v}_i$ and weights w_i by solving Eq. (26). The values are shown in Table 3.

References

1. Öttinger, H.C., *Beyond Equilibrium Thermodynamics*, Wiley, 2005.
2. Zhenhua Chai and Baochang Shi, Applied Mathematical Modelling **32**, 2050 (2008).
3. Jackson John David, *Electrodinámica clásica*, Editorial Alhambra S.A., first edition, 1966.

4. M. Ottaviani and F. Romanelli and R. Benzi and M. Briscolini and P. Santangelo and S. Succi, Physics of Fluids B: Plasma Physics **2**, 67 (1990).
5. Tomasi, J and Persico, M, Chemical Reviews **94**, 2027 (1994).
6. Honig, B and Nicholls, A, Science **268**, 1144 (1995).
7. Baker, NA and Sept, D and Joseph, S and Holst, MJ and McCammon, JA, Proc. of the Nat. Acad. of Sci. **98**, 10037 (2001).
8. Marques, M.A.L. and Gross, E.K.U., Annual Review of Physical Chemistry **55**, 427 (2004), PMID: 15117259.
9. Sheng Meng and Efthimios Kaxiras, The Journal of Chemical Physics **129**, 054110 (2008).
10. Gross, E.K.U. and Dreizler, R.M., *Density Functional Theory*, NATO ASI Series: Physics, Springer, 1995.
11. R. Benzi and S. Succi and Vergassola, Phys. Rep. **222**, 145 (1992).
12. S. Chen and G. Doolen, Annu. Rev. Fluid Mech. **30**, 329 (1998).
13. Chen, Shiyi and Doolen, Gary D., Annual Review of Fluid Mechanics **30**, 329 (1998).
14. Ponce Dawson, S. and Chen, S. and Doolen, G. D., The Journal of Chemical Physics **98**, 1514 (1993).
15. Mendoza, M. and Muñoz, J. D., Phys. Rev. E **82**, 056708 (2010).
16. S. Succi and R. Benzi, Physica D **69 3-4**, 327 (1993).
17. Mendoza, M. and Boghosian, B. M. and Herrmann, H. J. and Succi, S., Phys. Rev. Lett. **105**, 014502 (2010).
18. Mendoza, M. and Boghosian, B. M. and Herrmann, H. J. and Succi, S., Phys. Rev. D **82**, 105008 (2010).
19. Guo, Zhaoli and Zhao, T. S. and Shi, Yong, The Journal of Chemical Physics **122**, (2005).
20. Jinku Wang and Moran Wang and Zhixin Li, Journal of Colloid and Interface Science **296**, 729 (2006).
21. Miki Hirabayashi and Yu Chen and Hirotada Ohashi, JSME International Journal Series B Fluids and Thermal Engineering **44**, 45 (2001).
22. Xiaoyi He and Ning Li, Computer Physics Communications **129**, 158 (2000).
23. Baochang Shi and Bin Deng and Rui Du and Xingwang Chen, Computers and Mathematics with Applications **55**, 1568 (2008).
24. Chikatamarla, Shyam S. and Karlin, Iliya V., Phys. Rev. E **79**, 046701 (2009).
25. Mendoza, M. and Karlin, I. and Succi, S. and Herrmann, H. J., Phys. Rev. D **87**, 065027 (2013).
26. Mendoza, M. and Succi, S. and Herrmann, H. J., Phys. Rev. Lett. **113**, 096402 (2014).
27. Chapman, S. and Cowling, T.G., *The Mathematical Theory of Non-uniform Gases: An Account of the Kinetic Theory of Viscosity, Thermal Conduction and Diffusion in Gases*, Cambridge Mathematical Library, Cambridge University Press, 1970.
28. Bhatnagar, P. L. and Gross, E. P. and Krook, M., Phys. Rev. **94**, 511 (1954).
29. Grad, Harold, Communications on Pure and Applied Mathematics **2**, 331 (1949).
30. Sumesh P. Thampi and Santosh Ansumali and R. Adhikari and Sauro Succi, Journal of Computational Physics **234**, 1 (2013).
31. Rashmi Ramadugu and Sumesh P. Thampi and Ronojoy Adhikari and Sauro Succi and Santosh Ansumali, EPL (Europhysics Letters) **101**, 50006 (2013).
32. Hager, W.W., *Applied numerical linear algebra*, Prentice Hall PTR, 1988.
33. Briggs, W.L. and Henson, V.E. and McCormick, S.F., *A Multigrid Tutorial: Second Edition*, Society for Industrial and Applied Mathematics, 2000.